

Google App Script

O que é?

O Google App Script (GAS) é uma plataforma de **desenvolvimento de aplicativos**, associados aos produtos da Google ou de terceiros. Assim, o desenvolvedor pode **utilizar diversos recursos do Google Workspace** e integrar diversas APIs em seu programa.



Sua **sintaxe é similar ao Java**, sendo assim, sua linguagem é baseada em

[Programação Orientada a Objetos](#).

O GAS fornece diversos métodos para os objetos que você queira usar, e você pode [acessar a documentação](#) para conferir as possibilidades de desenvolvimento do seu programa:

Método	Tipo de retorno	Breve descrição
<code>copyBlob()</code>	Blob	Retorna uma cópia desse blob.
<code>getAs(contentType)</code>	Blob	Retorna os dados dentro desse objeto como um blob convertido para o tipo de conteúdo especificado.
<code>getBytes()</code>	Byte[]	Recebe os dados armazenados nesse blob.
<code>getContentType()</code>	String	Recebe o tipo de conteúdo dos bytes neste blob.
<code>getDataAsString()</code>	String	Recebe os dados desse blob como uma string com codificação UTF-8.
<code>getDataAsString(charset)</code>	String	Recebe os dados desse blob como uma string com a codificação especificada.
<code>getHash()</code>	String	Recebe o hash de conteúdo SHA1 para este anexo.
<code>getName()</code>	String	Recebe o nome desse blob.
<code>getSize()</code>	Integer	Recebe o tamanho desse anexo.
<code>isGoogleType()</code>	Boolean	Retorna se este blob é um arquivo do Google Workspace (Planilhas, Documentos etc.).
<code>setBytes(data)</code>	Blob	Define os dados armazenados neste blob.

Conferindo os métodos possíveis na documentação relacionados a anexo do Gmail

Seu editor é executado pelo lado de servidor, ou seja, **não é necessário instalar nada para usá-lo**. Para começar a usar somente é preciso uma conexão a internet e uma conta Gmail.

Com o GAS é possível:

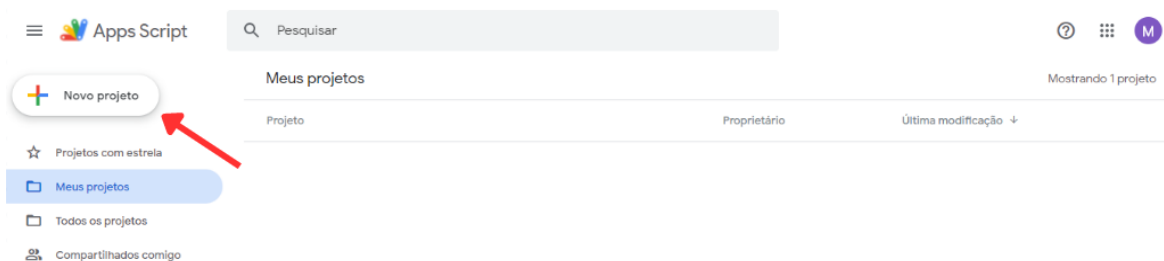
- Criar funcionalidade especiais para Google Sheets, GoogleDocs, Gmail e outros produtos relacionados
- Converter seu AppScript em um webApp
- Usar Cloud Computing para desenvolver programas sem baixar nada

[Acessar a home do Google App Script](#)

Guia Rápido

Os Básicos:

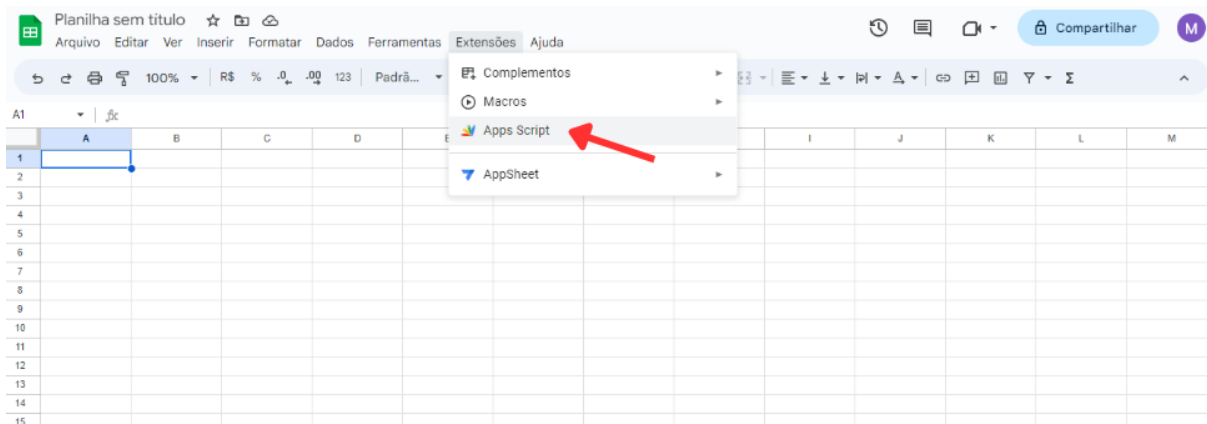
Para começar, [abra seu editor](#), e crie um novo projeto para começar a programar. Assim, você irá criar um [projeto StandAlone](#), ou autônomo. Isto é, um script independente e não vinculado com nenhum serviço do Google Workspace, e que aparecerá entre seus arquivos do Google Drive:



Caso queira criar um [projeto vinculado](#), você deve criá-lo a partir de um documento (Docs, Sheets, Apresentação, etc), seu arquivo anexado é armazenado no chamado "container".

“Os scripts vinculados geralmente se comportam como scripts independentes, mas não aparecem no Google Drive, não podem ser removidos do arquivo a que estão vinculados e recebem alguns privilégios especiais sobre o arquivo pai.”

Você poderá fazer isso , por exemplo, pelo Google Sheets, acessando Extensões e clicando Apps Script:

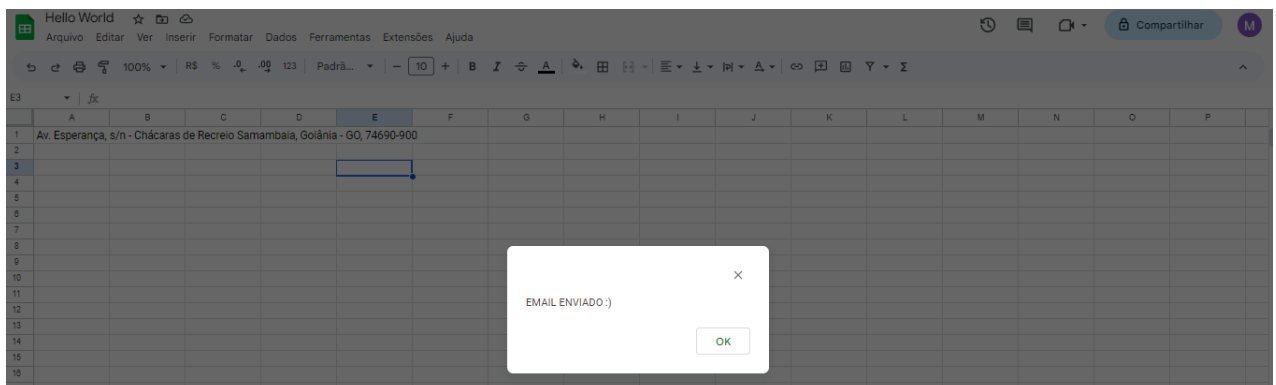


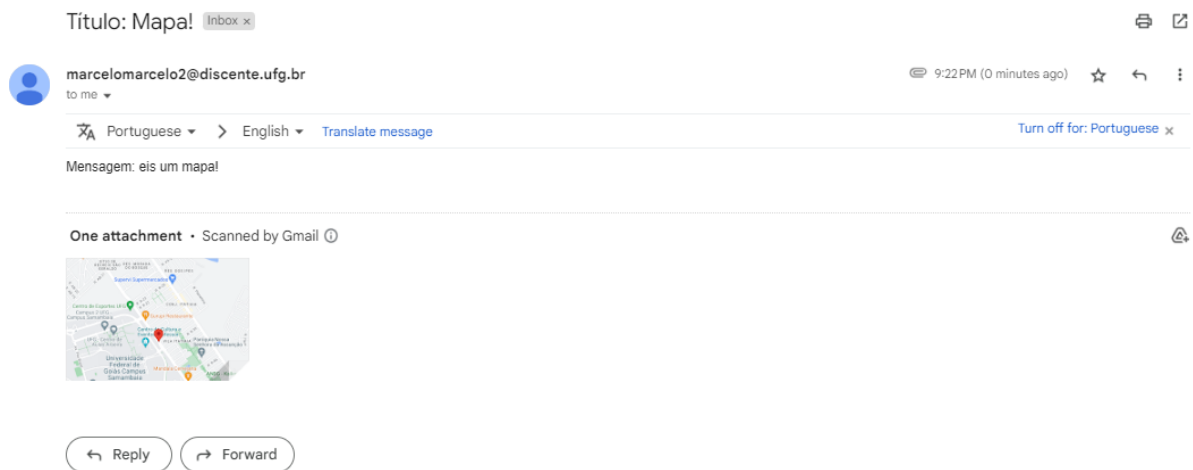
Meu primeiro Projeto:

A seguir, será mostrado como você poderá enviar mapas personalizados para emails. Primeiro, crie um script vinculado a uma planilha-como mostrado anteriormente-. Em seguir, siga o código abaixo:

```
function enviaMapa() {  
  // Armazena a planilha atual criada em uma variável  
  var planilha = SpreadsheetApp.getActiveSheet();  
  
  // Guarda o valor da primeira célula da planilha (A1)  
  var endereco = planilha.getRange('A1').getValue();  
  
  // Cria um mapa do GoogleMaps correspondente a esse endereço  
  var mapa = Maps.newStaticMap().addMarker(endereco)  
  
  // envia mapa para um email  
  GmailApp.sendEmail('marcelohlferreira@gmail.com',  
    'Título: Mapa!',  
    'Mensagem: eis um mapa!',  
    {attachments: [mapa]})  
};  
  
// Enviar Pop Up na interface do usuário (UI) ao abrir a planilha novamente  
// Esse tipo de método é exclusivo a BoundScripts/ vinculados  
SpreadsheetApp.getUi().alert("EMAIL ENVIADO :)");  
}
```

O código mostra os variados métodos de diferentes serviços que você pode integrar em seus apps no GAS. Após salvar seu código e executar a função enviaMapa(), você receberá uma mensagem de confirmação de envio na planilha criada, assim como um email com um Mapa!





Limitações

→ O tempo de execução máximo suportado por script é de 6 minutos, e [30 minutos](#) em quaisquer contas empresariais pagas ou contas educacionais. Para contornar isso você pode:

- ◆ Pesquisar maneiras de tornar seu código mais eficiente,
- ◆ Buscar métodos que permitam você executar seu script várias vezes, até que todo seu script todo seja executado.

[Exemplo](#)

→ Existe uma série de [Cotas e Limites de uso para os serviços Google](#), são algumas delas:

	Conta Legado (Gratuito)	Conta Workspace(Pago)
Slides criados	250 / dia	1.500 / dia
Planilhas criadas	250 / dia	3.200 / dia
Execuções simultâneas	30 / usuário	30 / usuário
Anexos de e-mails	250 / msg	250 / msg
Tamanho do corpo do e-mail	200 KB / msg	400 KB / msg
Destinatários de e-mail por mensagem	50 / msg	50 / msg

Tamanho total dos anexos do e-mail	25 MB / msg	25 MB / msg
Documentos criados	250 / dia	1.500 / dia
Arquivos convertidos	2.000 / dia	4.000 / dia
Destinatários de e-mail por dia	100* / dia	1.500* / dia
Destinatários de e-mail por dia no domínio	100* / dia	2.000 / dia
Leitura/gravação de e-mail (excluindo enviar)	20.000 / dia	50.000 / dia

→